APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/066,479	01/30/2002	Jim Pruyne	10006791-1	3529

7590   05/30/2008

HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P. O. Box 272400
Fort Collins, CO 80527-2400

EXAMINER
NAWAZ, ASAD M

ART UNIT	PAPER NUMBER
2155	

MAIL DATE	DELIVERY MODE
05/30/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

_____

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

_____

Ex parte JIM PRUYNE and SVEND FROLUND

_____

Appeal 2008-0579
Application 10/066,479
Technology Center 2100

_____

Decided: May 30, 2008

_____

Before JAMES D. THOMAS, JEAN R. HOMERE, and STEPHEN C. SIU,
Administrative Patent Judges.

SIU, *Administrative Patent Judge.*

DECISION ON APPEAL

I. STATEMENT OF THE CASE

Appellants appeal under 35 U.S.C. § 134(a) from the Examiner's
Final Rejection of claims 24-45. We have jurisdiction under 35 U.S.C.
§ 6(b). We affirm.

A. INVENTION

The invention at issue involves error handling for web-based transaction processing (Spec. 5). In particular, a server generates an identifier corresponding to a transaction and provides the identifier in a form to a client. The client fills out and posts the form to the server. In response, the server generates a status page to inform the user of the status of the transaction. If failures occur, the error handling mechanism provides exactly-once error handling semantics (*id.*).

B. ILLUSTRATIVE CLAIM

Claim 24, which further illustrates the invention, follows:

24.   A method for performing a web transaction, comprising:

obtaining a form that includes a unique identifier for the web transaction;

initiating a database update and generating a log for the database update such that the log is identified by the unique identifier;

obtaining a request to reload a status page such that the request includes the unique identifier;

accessing the log in response to the request and retrying the database update if the log indicates a failure of the database update such that the database update is performed at most once.

2

C. REJECTION

Claims 24-45 stand rejected under 35 U.S.C. § 102(e) as being
anticipated by U.S. Patent No. 6,259,701 ("Shur"). Claims 1-23 have been
cancelled.

II. CLAIM GROUPING

> "When multiple claims subject to the same ground of rejection are
> argued as a group by appellant, the Board may select a single claim
> from the group of claims that are argued together to decide the appeal
> with respect to the group of claims as to the ground of rejection on the
> basis of the selected claim alone. Notwithstanding any other
> provision of this paragraph, the failure of appellant to separately argue
> claims which appellant has grouped together shall constitute a waiver
> of any argument that the Board must consider the patentability of any
> grouped claim separately." 37 C.F.R. § 41.37(c)(1)(vii) (2006).[1]

Appellants argue claims 24, 27, 29, 32, 35, 39, and 42 as a first group
(App. Br. 10-14); claims 25, 33, and 40 as a second group (App. Br. 14);
claims 26, 34, and 41 as a third group (App. Br. 14); claims 28, 36, and 43 as
a fourth group (App. Br. 14); claims 30, 37, and 44 as a fifth group (App. Br.
15); and claims 31, 38, and 45 as a sixth group (App. Br.15). We select
claim 24 as the sole claim on which to decide the appeal of the first group,
claim 25 as the sole claim on which to decide the appeal of the second
group, claim 26 as the sole claim on which to decide the appeal of the third

---

[1] We cite to the version of the Code of Federal Regulations in effect at the
time of the Appeal Brief. The current version includes the same rules.

group, claim 28 as the sole claim on which to decide the appeal of the fourth group, claim 30 as the sole claim on which to decide the appeal of the fifth group, and claim 31 as the sole claim on which to decide the appeal of the sixth group.

III. CLAIMS 24, 27, 29, 32, 35, 39, AND 42

The Examiner finds that Shur discloses each feature of claim 24 at col. 4, ll. 41-64; col. 5, ll. 20 – col. 6, l. 3; and col. 6, ll. 10-41 (Ans. 3-4). Shur discloses a "user of a client" requesting "a copy of the page containing the Multicast sessions" (col. 5, ll. 32-33) where the page contains "a list of URLs describing the sessions" (col. 5, l. 31). After authentication of the client/user, the "server 206 returns a page to the client containing a list of sessions" (col. 5, ll. 36-37), from which the client/user may choose. The server also "returns a form to the client requesting the client to enter a login id and password appropriate for creating a new session" (col. 5, ll. 45-46). After further authentication, the server "returns . . . a form to the client with fields to be filled in by the user for the session name, session description . . ." (col. 5, ll. 50-52). The form, having been filled out by the user, "is returned to the server **206**" (col. 5, l. 59). "[A]n error message is returned to the client" if key fields are incorrectly entered (col. 5, l. 61) but otherwise stored "in a file that is indexed to the login id used to access the form" (col. 5, ll. 62-63). The data stored in the file is "then made available to a Session Announcement Protocol (SAP process 210 . . . (that) . . . periodically

announces the session onto a well-known Multicast IP address and port"
(col. 5, ll. 63-67).

Appellants argue that although "the client in Shur receives a list of
Multicast sessions," "the client in Shur never obtains 'a form' as recited in
claim 24" because "[a] list of sessions is not a 'form'" (App. Br. 11). As set
forth above, Shur discloses that the server "returns . . . a form to the client
with fields to be filled in by the user for the session name, session
description . . ." (col. 5, ll. 50-52). This form is further "returned to the
server **206**" (col. 5, l. 59). Using a broad but reasonable interpretation, we
find that Shur's explicit disclosure of a "form" is equivalent to the "form"
recited in claim 24. Therefore, we are unconvinced by Appellants'
argument. "[T]he PTO gives claims their 'broadest reasonable
interpretation.'" *In re Bigio*, 381 F.3d 1320, 1324 (Fed. Cir. 2004) (quoting
In re Hyatt, 211 F.3d 1367, 1372 (Fed. Cir. 2000)).

Appellants also argue that "Shur does not even suggest that this list
(of Multicast sessions) includes the login ID and password or any unique
identifier required for the web transaction" (App. Br. 11-12). The issue is
whether Shur discloses a form that includes a unique identifier for a web
transaction as recited in claim 24. Shur discloses that the user/client receives
a "form" "with fields to be filled in by the user" which includes, for
example, "session name, session description, a URL address for more
detailed or related information, . . ." (col. 5, ll. 50-53). We find that the
client/user can input any of the session name, session description, or URL

address into the form. Because the session name, session description, URL, etc. identify a transaction by identifying a corresponding session, we find that the unique identifier for the web transaction recited in claim 24 is equivalent to any of the session name, session description, URL, etc. The server obtains the form (including the unique identifier(s)) when the form is "returned to server **206**" (col. 5, l. 59).

Appellants argue that although "Shur sends a client an error message," "an error message is not a status page . . . (and) is not 'reloaded'" as recited in claim 24 (App. Br. 12). Shur discloses that the "server 206 checks that certain key fields are correct and filled in . . . if not, an error message is returned to the client" (col. 5, ll. 59-61). Using a broad but reasonable interpretation, we construe the term "status page" to include a page or form that indicates a present condition of an entity. Because the error message of Shur provides the user with an indication of the present condition of the user request such as the presence of an error that needs to be corrected, we disagree with Appellants that "an error message is not a status page."

As described above, the user responds to the "status page" (i.e., error message) by providing the corrected information in the form. Using a broad but reasonable interpretation, we construe the term "reload" to include putting something into or onto a subsequent time. The form of Shur is initially provided to the user "to be filled in by the user for the session name, session description, a URL address . . ." (col. 5, ll. 50-51) and subsequently "returned to the server **206**" (col. 5, l. 59). After an error or omission is

detected in the form at the server, "an error message is returned to the client (col. 5, l. 61). In order for the user/client to provide the corrected information, the form must be "reloaded" (i.e., re-posted to the user for re-entry of data). Thus, we find that Shur discloses "reloading" the form.

Alternatively, the server detects an error in the user/client's data entry in the form and sends an error message as a request for corrected information to the user/client (as described above). The user/client receives the request for corrected information and re-enters the data into the form. Under this interpretation, the user obtains a request from the server to reload the form with the corrected information. In addition, the request from the server identifies the erroneous form such that the erroneous transaction is uniquely identified. Under this alternate broad but reasonable interpretation, Shur discloses obtaining a request to reload a "status page" that includes a unique identifier as recited in claim 24.

Appellants argue that "Shur sends a client an error if the form is not correctly completed" but does not disclose "that the error message includes the login ID of the client" (App. Br. 12). We note that claim 24 recites the request includes the unique identifier but does not recite that the request includes "the login ID of the client." Even assuming that Shur does not disclose an error message that includes the login ID of the client, as Appellants assert, we find the alleged lack of disclosure to be immaterial because claim 24 does require that an error message include the login ID of a client.

Appellants further argue that "Shur . . . never teaches a request to reload a status page . . . (and) never teaches accessing the file (i.e., log) in response to a request to reload a status page" (App. Br. 13) and that Shur also fails to disclose "retrying an update if the log itself indicates a failure. The log in Shur never indicates a failure" (App. Br. 13). As set forth above, Shur discloses the server receiving a form from a client containing "the session name, session description . . ." (col. 5, l. 51-52) (i.e., a unique identifier) and initiating storage of session data "in a file that is indexed to the login id used to access the form" (col. 5, ll. 62-63) (i.e., "a log"). If "certain key fields" are not "correct and filled in" (col. 5, l. 60), the server sends a request for corrected information (i.e., an error message requesting that the client reload the corrected information in the form) to the client. Under a broad but reasonable interpretation, we find that "a log" includes any record of events. The form and error message of Shur includes a record of events (i.e., a record of entry of erroneous data, notification of errors to the user/client, and request for re-entry of corrected data) and therefore constitutes a "log." In response to accessing the "log" (i.e., accessing the error message and form, the error message indicating a failure of the transaction due to erroneous information), the user/client system then retries the transaction (because the "log" indicates previous failure of the transaction due to erroneous information).

Appellants argue that "Shur never teaches whatsoever that if the log indicates a failure, then the database update is performed at most once"

(App. Br. 14). Shur also discloses that if "certain key fields are correct and filled in" (col. 5, l. 60), the "server **206** stores the data in the file that is indexed to the login id used to access the form" (col. 5, ll. 61-63). The data is stored in the file "at most once" because the data is only stored after all "key fields are correct and filled in" (col. 5, l. 60). Therefore, we find that Shur discloses each feature of claim 24.

Because Appellants have failed to demonstrate that the Examiner erred in rejecting claim 24, we affirm the rejection of claim 24 and of claims 27, 29, 32, 35, 39, 42, which fall therewith.

IV. CLAIMS 25, 33, AND 40

Appellants argue that Shur "is completely silent about sending the form as a 'post command'" (App. Br. 14).

Under a broad but reasonable interpretation, we adopt the ordinary and customary meaning of the term "post command" to include any command that routes a message to a correct destination by identifying a recipient's location, performing header operations, and/or providing a file to be posted. Shur discloses that the server "returns . . . a form to the client with fields to be filled in by the user" (col. 5, ll. 50-51) and "the user fills in the fields" and "the filled-in form is returned to the server **206**" (col. 5, ll. 58-59). Because Shur discloses that a form is sent between a server and client and that the form is sent from/received at the proper destination, we find that Shur discloses commands that route a message (or form) to correct

destinations. As such, we find that this disclosure is equivalent to the "post command" recited in claim 25.

Even assuming that Shur does not disclose the exact term "post command," as Appellants assert, anticipation "is not an 'ipsissimis verbis' test." *In re Bond*, 910 F.2d 831, 832 (Fed. Cir. 1990) (citing *Akzo N.V. v. United States Int'l Trade Comm'n*, 808 F.2d 1471, 1479 (Fed. Cir. 1986)). "An anticipatory reference . . . need not duplicate word for word what is in the claims." *Standard Havens Prods. v. Gencor Indus.*, 953 F.2d 1360, 1369 (Fed. Cir. 1991). Therefore, whether Shur discloses the term "post command" or not is immaterial.

It follows that Appellants have failed to demonstrate that the Examiner erred in rejecting claim 25. Therefore, we affirm the rejection of claim 25 and of claims 33 and 40, which fall therewith.

V. CLAIMS 26, 34, AND 41

Appellants argue that "nowhere does Shur teach that the error message automatically generates a request to reload at the client" (App. Br. 14).

As set forth above, we broadly but reasonably construe the term "status page" to include a page or form that indicates a present condition of an entity. Shur discloses "a form . . . with fields to be filled in by the user" (col. 5, ll. 50-51). "When the user fills in the fields and selects a create button, the filled-in form is returned to the server 206" (col. 5, ll. 57-59). If

10

the form is not correctly filled in, "an error message is returned to the client" (col. 5, ll. 60-61). Because the form and/or error message provides information of the present condition of the transaction (e.g., in progress, corrections needed, etc.), we find that the form and/or error message of Shur constitutes the "status page" recited in claim 26. In addition, the server, upon receiving the form containing erroneous information from the client, automatically returns an error page prompting the user/client to re-enter corrected information. As such, the "request to reload" (i.e., the server request for corrected information from the client) is automatically generated by receipt at the server of the erroneous form (i.e., the "status page" from the client).

It follows that Appellants have failed to demonstrate that the Examiner erred in rejecting claim 26. Therefore, we affirm the rejection of claim 26 and of claims 34 and 41, which fall therewith.

VI. CLAIMS 28, 36, AND 43

Appellants argue that "[i]n Shur, the numerous fields and session ids are not included in a request to reload" (App. Br. 14).

As previously discussed, Shur discloses a request from a server to re-enter corrected information after the server receives a form that contains "certain key fields" that are not "correct and filled in" (col. 5, l. 60) from the user/client and "stores the data in a file" (col. 5, ll. 62) after corrected data is received in the form from the user/client. We find that the request to reload

(i.e., the request for corrected information in the form), although including "certain key fields" that are incorrect, includes other fields with initially correct information. Because the server contains "a set of data" such as

> "session name, session description, a URL address for more detailed or related information, a field to set the Multicast packet Time to Live value (TTL), selection boxes for various media tools, and associated coding formats, Multicast addresses and ports to use, the name and phone number of a responsible person, and the duration of the session announcement" (col. 5, ll. 51-57),

some of which are correctly filled into the form, we find that the request to reload from the server already contains "data for retrying the database update" (i.e., correctly filled in data from the user/client).

It follows that Appellants have failed to demonstrate that the Examiner erred in rejecting claim 28. Therefore, we affirm the rejection of claim 28 and of claims 36 and 43, which fall therewith.

VII. CLAIMS 30, 37, AND 44

Appellants argue that Shur fails to "teach 'rolling back the database update' or performing a roll back after a 'timeout period'" (App. Br. 15).

The Examiner finds that rolling back a database "is a basic security feature that is well known in the art especially in the case when sessions timeout" (Ans. 8). Appellants do not refute this finding. Also, the Examiner finds that Shur discloses "a timeout mechanism . . . (col 5, lines 16-20 and col 6, lines 1-3)" (*id.*). Shur discloses "a field to set the Multicast packet

Time to Live value (TTL)" (col. 5, ll. 53-54). Because rolling back a database "is a basic security feature that is well known in the art" and Shur discloses a "time out mechanism" that includes a user indicating a "Time to Live value (TTL)," we find that Shur performs a well known security measure during session timeouts (i.e., rolling back the database) with the timeout period as indicated by the user in the "Time to Live value (TTL)" in the form.

It follows that Appellants have failed to demonstrate that the Examiner erred in rejecting claim 30. Therefore, we affirm the rejection of claim 30 and of claims 37 and 44, which fall therewith.

VIII. CLAIMS 31, 38, AND 45

Appellants argue that Shur does not disclose "determining a timeout period in response to a **timestamp** contained in the status page" (App. Br. 15).

We broadly but reasonably construe the term "timestamp" to include any indicator of time. As set forth above, Shur discloses a timeout period as specified by the user/client in the "Time to Live value (TTL)" field of the form. Because Shur discloses timing out after a specified period of time, Shur must also monitor the current time in order to be able to time out at the proper moment. In order to keep track of time, Shur must utilize "an indicator of time." Because we find that a timestamp constitutes "an indicator of time," we find that Shur must include a "timestamp" to keep

track of time. As such, we agree with the Examiner that Shur discloses the
timestamp in the "status page" (i.e., the form that includes the "TTL" field).

It follows that Appellants have failed to demonstrate that the
Examiner erred in rejecting claim 31. Therefore, we affirm the rejection of
claim 31 and of claims 38 and 45, which fall therewith.

VII. ORDER

In summary, the rejection of claims 24-45 under § 102(e) is affirmed.

No time period for taking any subsequent action in connection with
this appeal may be extended under 37 C.F.R. § 1.136(a).

AFFIRMED

rwk

HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P.O. Box 272400
Fort Collins, CO 80527-2400